

Golo

PETIT MAIS COSTAUD



#gololang #rivierajug

Merci





Philippe Charrière | **@k33g_org**

Responsable Marketing

(Golo Dancer & Committer)

@Typeunsafe



Philippe Charrière | @k33g_org

Responsable Marketing

(Golo Dancer & Committer)

@Typeunsafe

Directeur Technique



#gololang #rivierajug

Nous parlerons de ...

Golo, quoi, qui, pourquoi, ...

(au fait, c'est un langage)

Golo in action, particularités,  Java

(du code, du code, du code)

Hacking Golo

(le langage dont vous êtes le héros)

...

Golo?

Quoi, Qui, Pourquoi, ...

Golo?

1 langage **dynamique** pour la **JVM**
à base **d'Invokedynamic**
facile à utiliser
facile à modifier

Golo?

```
module hello
```

```
function hello = |who| {  
  println("Hello " + who)  
}
```

```
function main = |args| {
```

```
  let salut = |qui| -> println("Salut " + qui)
```

```
  salut("John")  
  hello("Bob")  
}
```


Golo?

Créé par **@jponge** (aka “le doc”)
Boss de TypeUnsafe



Golo?

Dynamid
Middlewares **Dynamiques**
Expertise Invokedynamic

Golo?

Middlewares Dynamiques?

Golo? ... à savoir

Léger 530 kb

Golo <3 **Java**

Golo est **rapide**

dans un contexte dynamique **#notroll**

Golo?

Benchs

ceci n'est pas un #troll



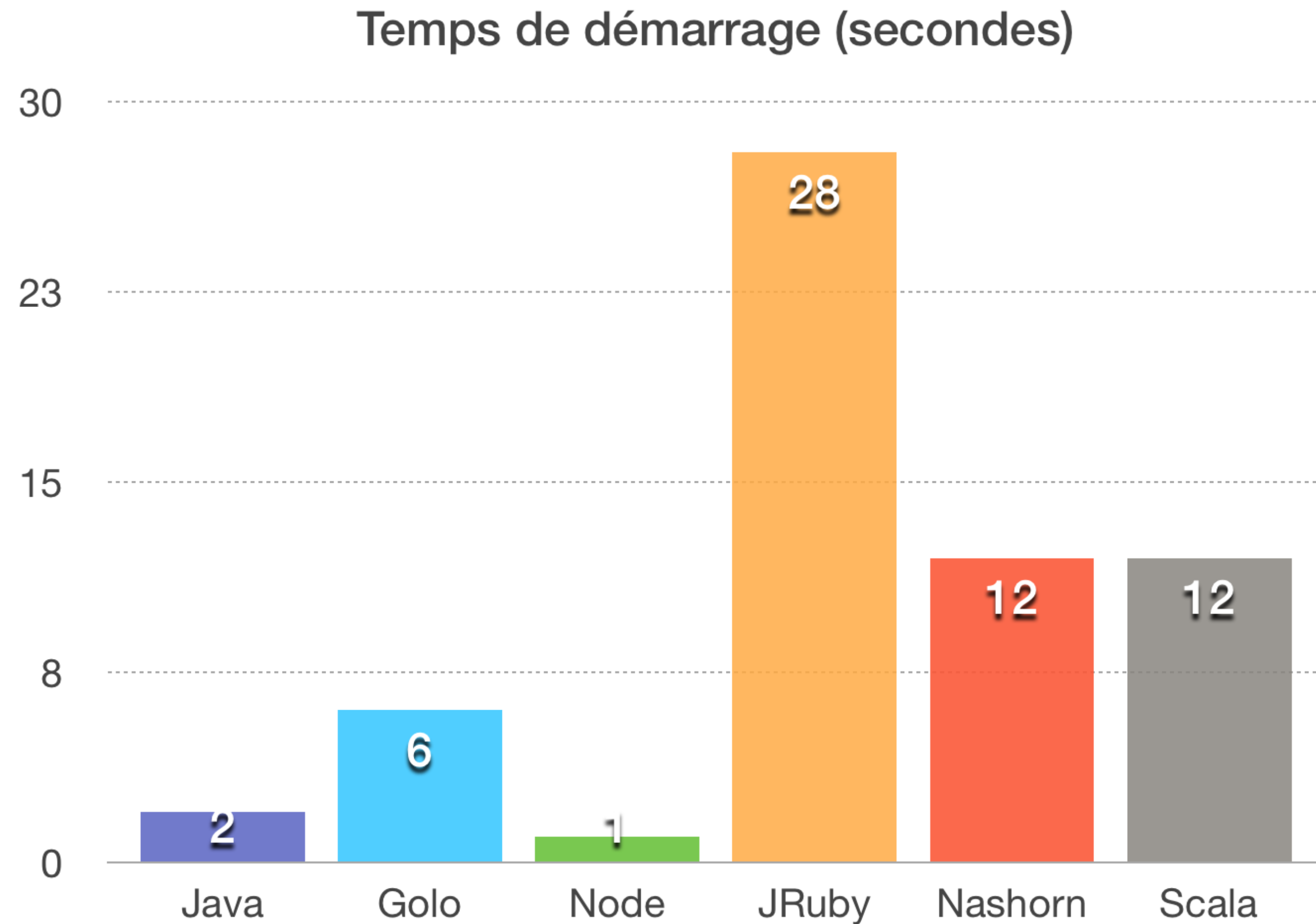
GoLoBox

#gololang #rivierajug

Golo, quoi, qui, pourquoi, ...

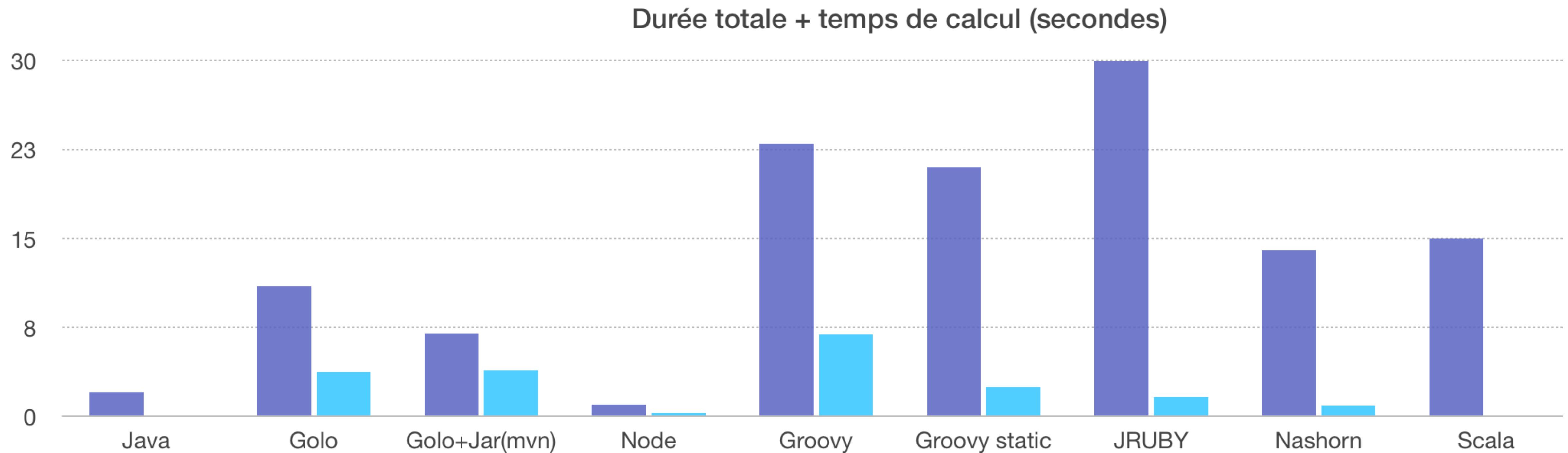
Temps de démarrage - plateforme : RPI B+ / Java 8

Un programme simple: afficher "Hello"



Approche de PI - plateforme : RPI B+ / Java 8

Durée totale du traitement (lancement + calcul) et temps de calcul (compris dans la durée de traitement)



Conclusion

Java + dynamique + délais courts +
mémoire (**IOT**)
-> Golo

Pourquoi faire du Golo?

pour le **fun**

pour **apprendre** (pas que Golo)

pour **expérimenter** & prototyper

pour “**partager**” & participer

pour de l’**embarqué**

pour avoir son propre langage

...

Golo? ... à savoir

Un langage développé
sur le temps libre,
la bonne volonté
et la passion
... et en peu de temps

Golo in action

Particularités,  Java

Préambule

Générer un projet Golo

Simple: `golo new hello`

Maven: `golo new hellomvn --type maven`

Gradle: `golo new hellogrd --type gradle`

Démos

bouts de code

Il y a aussi ...

tout ce dont je n'ai pas eu le temps de parler

<http://golo-lang.org/documentation/next/>

Applications

du web, des outils, ...

Créer des “webapps” rapidement

qui “parlent” avec Java, légères

Créer des utilitaires

générateurs de code (à la Yeoman)

etc. ...



Benchs Bis

#gololang #rivierajug

Applications


```
class BasicSimulation extends Simulation {  
  
  val httpConf = http  
    .baseUrl("http://192.168.0.15:8080")  
  
  val scn = scenario("Add Humans").repeat(10) {  
    exec(  
      http("add human")  
        .post("/humans")  
        .body(StringBody("{\"firstName\":\"John\",\"lastName\":\"Doe\"}"))  
        .asJSON  
        .check(status.is(200))  
    )  
  }  
  
  setUp(scn  
    .inject(rampUsers(20) over (60)).protocols(httpConf))  
}
```

Golo + SparkJava

STATISTICS Expand all groups Collapse all groups											
Requests ↕	🔄 Executions				🕒 Response Time (ms)						
	Total ↕	OK ↕	KO ↕	% KO ↕	Min ↕	Max ↗	Mean ↕	Std Dev ↕	95th pct ↕	99th pct ↕	Req/s ↕
Global Information	200	200	0	0 %	8	126	13	9	23	32	3.50
add human	200	200	0	0 %	8	126	13	9	23	32	3.50

SparkJava only

STATISTICS Expand all groups Collapse all groups											
Requests ↗	🔄 Executions				🕒 Response Time (ms)						
	Total ↕	OK ↕	KO ↕	% KO ↕	Min ↕	Max ↕	Mean ↕	Std Dev ↕	95th pct ↕	99th pct ↕	Req/s ↕
Global Information	200	200	0	0 %	7	50	11	5	20	24	3.50
add human	200	200	0	0 %	7	50	11	5	20	24	3.50

```
class BasicSimulationGetAll extends Simulation {  
  val httpConf = http  
    .baseUrl("http://192.168.0.15:8080")  
  
  val scn = scenario("Get All Humans").repeat(10) {  
    exec(  
      http("get humans")  
        .get("/humans")  
        .check(status.is(200))  
    )  
  }  
  
  setUp(scn  
    .inject(rampUsers(20) over (60)).protocols(httpConf))  
}
```


Golo + SparkJava

▶ STATISTICS

Expand all groups | Collapse all groups

Requests ^	🔄 Executions				🕒 Response Time (ms)						
	Total ↕	OK ↕	KO ↕	% KO ↕	Min ↕	Max ↕	Mean ↕	Std Dev ↕	95th pct ↕	99th pct ↕	Req/s ↕
	Global Information	200	200	0	0 %	64	193	81	16	103	154
get humans	200	200	0	0 %	64	193	81	16	103	154	3.46

SparkJava only

▶ STATISTICS

Expand all groups | Collapse all groups

Requests ^	🔄 Executions				🕒 Response Time (ms)						
	Total ↕	OK ↕	KO ↕	% KO ↕	Min ↕	Max ↕	Mean ↕	Std Dev ↕	95th pct ↕	99th pct ↕	Req/s ↕
	Global Information	200	200	0	0 %	37	174	50	15	74	98
get humans	200	200	0	0 %	37	174	50	15	74	98	3.48

```
class BasicSimulationGetAll extends Simulation {  
  val httpConf = http  
    .baseUrl("http://192.168.0.15:8080")  
  
  val scn = scenario("Get All Humans").repeat(10) {  
    exec(  
      http("get humans")  
        .get("/humans")  
        .check(status.is(200))  
    )  
  }  
  
  setUp(scn  
    .inject(rampUsers(200) over (60)).protocols(httpConf))  
}
```

Golo + SparkJava

STATISTICS Expand all groups Collapse all groups											
Requests ^	Executions				Response Time (ms)						
	Total ↕	OK ↕	KO ↕	% KO ↕	Min ↕	Max ↕	Mean ↕	Std Dev ↕	95th pct ↕	99th pct ↕	Req/s ↕
Global Information	2000	2000	0	0 %	68	46862	12550	6998	26178	31191	10.12
get humans	2000	2000	0	0 %	68	46862	12550	6998	26178	31191	10.12

SparkJava only

STATISTICS Expand all groups Collapse all groups											
Requests ^	Executions				Response Time (ms)						
	Total ↕	OK ↕	KO ↕	% KO ↕	Min ↕	Max ↕	Mean ↕	Std Dev ↕	95th pct ↕	99th pct ↕	Req/s ↕
Global Information	2000	2000	0	0 %	40	15208	4685	2795	9879	11830	18.76
get humans	2000	2000	0	0 %	40	15208	4685	2795	9879	11830	18.76

Hacking Golo

Facile!

“Augmenter” Golo en **Java** ... et en **Golo**

Univers Golo

Vers l'infini & au-delà



jponge
Julien Ponge

Communauté



Artpej
sylvain Desgrais



yloiseau
Yannick Loiseau



erdi
Marcin Erdmann



mojavelinux
Dan Allen



ath0mas
Alexis THOMAS



flemouel
Frédéric Le Mouël



davidfestal
David Festal



nstouls



rlespinasse
Romain Lespinasse



danielpetisme
Daniel PETISME



jeffmaury
Jeff MAURY



kelcecil
Kel Cecil



titimoby
Thierry



franckverrot
Franck Verrot



Mogztter
Guillaume Grossetie



k33g
Philippe CHARRIERE

Mention spéciale

@NinjaSquad

@SerliFr

@HGomez

Et tous les autres: testeurs, conférences, JUG, ...

<https://github.com/golo-lang/golo-lang#contributing>

The End

Merci à vous, Questions?, ...



Liens

Golo: <http://golo-lang.org/>
[@golo_lang](https://github.com/golo-lang/golo-lang)

Dynamid: <http://dynamid.citi-lab.fr/>

Citi-Lab: <http://www.citi-lab.fr/>

Contacts

Boss: @jponge

Marketing: @k33g_org | ph.charriere@gmail.com